Gradient Calculations for a General RNN

1. Gradients We Need to Compute

In a Recurrent Neural Network (RNN), the following gradients are typically computed:

(A) Output Layer Parameters

1. **Weights** (W_y): Gradient of the loss with respect to the output weights. 2. **Biases** (b_y): Gradient of the loss with respect to the output biases.

(B) Hidden Layer Parameters

3. **Input-to-Hidden Weights** (W_x): Gradient of the loss with respect to input weights. 4. **Hidden-to-Hidden Weights** (W_h): Gradient of the loss with respect to recurrent weights. 5. **Biases** (b_h): Gradient of the loss with respect to the hidden layer biases.

(C) Intermediate States

6. **Hidden States** (h_t): Gradient of the loss with respect to each hidden state. 7. **Outputs** ($\hat{y}_t$): Gradient of the loss with respect to the outputs.

2. General Definitions

(A) Output at Each Time Step

At each time step t, the output is computed as:

$$\hat{y}_t = g(W_y h_t + b_y)$$

Where:

- g: Output activation function (e.g., softmax, sigmoid, identity).
- W_y: Output weight matrix.
- b_y: Output bias vector.

(B) Hidden State Update

The hidden state at time t is updated as:

$$h_t = f(W_h h_{t-1} + W_x x_t + b_h)$$

Where:

- f: Hidden activation function (e.g., tanh, ReLU, sigmoid).
- W_h: Hidden-to-hidden weight matrix.
- W_x: Input-to-hidden weight matrix.
- b_h: Hidden layer bias vector.

(C) Loss Function

For a general loss function $\mathcal{L}$:

$$\mathcal{L} = \sum_{t=1}^{T} \mathcal{L}_t(\hat{y}_t, y_t)$$

Where $\mathcal{L}_t$ is the loss at time t (e.g., cross-entropy, MSE).

3. Step-by-Step Gradients

(A) Gradient of the Loss with Respect to the Output Layer Parameters

1. Gradient w.r.t. Output Weights (W_y)

$$\frac{\partial \mathcal{L}}{\partial W_y} = \sum_{t=1}^{T} \frac{\partial \mathcal{L}_t}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial W_y}$$

Breaking it down:

- $\frac{\partial \mathcal{L}_t}{\partial \hat{y}_t}$: Depends on the loss function $\mathcal{L}_t$. For example:
  - Cross-entropy: $\frac{\partial \mathcal{L}_t}{\partial \hat{y}_t} = \hat{y}_t - y_t$.
  - MSE: $\frac{\partial \mathcal{L}_t}{\partial \hat{y}_t} = 2(\hat{y}_t - y_t)$.
- $\frac{\partial \hat{y}_t}{\partial W_y} = h_t^\top$: The hidden state at time t.

Final expression:

$$\frac{\partial \mathcal{L}}{\partial W_y} = \sum_{t=1}^{T} \left(\frac{\partial \mathcal{L}_t}{\partial \hat{y}_t} \cdot h_t^\top \right)$$

2. Gradient w.r.t. Output Biases (b_y)

$$\frac{\partial \mathcal{L}}{\partial b_y} = \sum_{t=1}^{T} \frac{\partial \mathcal{L}_t}{\partial \hat{y}_t}$$

(B) Gradient of the Loss with Respect to the Hidden States

The gradient at hidden state h_t is recursive:

$$\frac{\partial \mathcal{L}}{\partial h_t} = \frac{\partial \mathcal{L}_t}{\partial h_t} + \frac{\partial \mathcal{L}}{\partial h_{t+1}} \cdot \frac{\partial h_{t+1}}{\partial h_t}$$

Breaking it down:

- **Direct Contribution:**

$$\frac{\partial \mathcal{L}_t}{\partial h_t} = \frac{\partial \mathcal{L}_t}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial h_t}$$

  - $\frac{\partial \hat{y}_t}{\partial h_t} = g'(W_y h_t + b_y) \cdot W_y$.
- **Recursive Contribution:**

$$\frac{\partial h_{t+1}}{\partial h_t} = f'(z_{t+1}) \cdot W_h$$

  - $f'(z_{t+1})$: Derivative of the hidden activation function f.

Final expression:

$$\frac{\partial \mathcal{L}}{\partial h_t} = \frac{\partial \mathcal{L}_t}{\partial h_t} + \left(\frac{\partial \mathcal{L}}{\partial h_{t+1}} \cdot f'(z_{t+1}) \cdot W_h \right)$$

(C) Gradient of the Loss with Respect to the Hidden Layer Parameters

3. Gradient w.r.t. Input-to-Hidden Weights (W_x)

$$\frac{\partial \mathcal{L}}{\partial W_x} = \sum_{t=1}^{T} \frac{\partial h_t}{\partial W_x} \cdot \frac{\partial \mathcal{L}}{\partial h_t}$$

Breaking it down:

- $\frac{\partial h_t}{\partial W_x} = f'(z_t) \cdot x_t^\top$.

  Final expression:

$$\frac{\partial \mathcal{L}}{\partial W_x} = \sum_{t=1}^{T} \left(f'(z_t) \cdot x_t^\top \cdot \frac{\partial \mathcal{L}}{\partial h_t} \right)$$

4. Gradient w.r.t. Hidden-to-Hidden Weights (W_h)

$$\frac{\partial \mathcal{L}}{\partial W_h} = \sum_{t=1}^{T} \frac{\partial h_t}{\partial W_h} \cdot \frac{\partial \mathcal{L}}{\partial h_t}$$

Breaking it down:

- $\frac{\partial h_t}{\partial W_h} = f'(z_t) \cdot h_{t-1}^\top$.

  Final expression:

$$\frac{\partial \mathcal{L}}{\partial W_h} = \sum_{t=1}^{T} \left(f'(z_t) \cdot h_{t-1}^\top \cdot \frac{\partial \mathcal{L}}{\partial h_t} \right)$$

5. Gradient w.r.t. Hidden Biases (b_h)

$$\frac{\partial \mathcal{L}}{\partial b_h} = \sum_{t=1}^{T} f'(z_t) \cdot \frac{\partial \mathcal{L}}{\partial h_t}$$

4. Multiplication Types: Matrix vs. Element-Wise

- **Element-Wise Multiplication:**
  - When applying the derivative of an activation function, e.g., $f'(z_t)$ or $g'(z_t)$.
  - Example: $f'(z_t) \cdot \frac{\partial \mathcal{L}}{\partial h_t}$.
- **Matrix Multiplication:**
  - When propagating gradients through weight matrices, e.g., W_h, W_x, W_y.
  - Example: $f'(z_t) \cdot W_h$.

5. Summary of Gradients

Parameter	Gradient Expression
W_y	$\sum_{t=1}^{T} \left(\frac{\partial \mathcal{L}_t}{\partial \hat{y}_t} \cdot h_t^\top \right)$
b_y	$\sum_{t=1}^{T} \frac{\partial \mathcal{L}_t}{\partial \hat{y}_t}$
W_x	$\sum_{t=1}^{T} \left(f'(z_t) \cdot x_t^\top \cdot \frac{\partial \mathcal{L}}{\partial h_t} \right)$
W_h	$\sum_{t=1}^{T} \left(f'(z_t) \cdot h_{t-1}^\top \cdot \frac{\partial \mathcal{L}}{\partial h_t} \right)$
b_h	$\sum_{t=1}^{T} f'(z_t) \cdot \frac{\partial \mathcal{L}}{\partial h_t}$